§ 2 Automation Device Systems and Structures

2.1 Automation Computers
2.2 Centralized and Decentralized Structures
2.3 Automation Hierarchies
2.4 Distributed Automation Systems
2.5 Automation Structures with Redundancy
Chapter 2 - Learning targets

- to know the different automation computers
- to know what is the special of the mode of operation of a PLC
- to be able to differ between centralized and decentralized structures
- to recognize combinations of automation system structures
- to know automation hierarchies and their requirements
- to understand what distributed automation systems are
- to know what the basic topologies of communication are
- to be able to differ between an open and a proprietary communication system
- to know what is meant by redundancy
- to know kinds of hardware redundancy and to be able to characterize them
- to be able to explain what is meant by diversity
§ 2 Automation Device Systems and Structures

2.1 Automation computers
2.2 Centralized and decentralized structures
2.3 Automation hierarchies
2.4 Distributed automation systems
2.5 Automation structures with redundancy
**Situation (1)**

- Programmable Logic Controllers since 1972
- Influenced by components and technologies
- Dependent on the task
- Beginning of the software

---

**Increase of the functionality and efficiency of the programmable logic controllers**

<table>
<thead>
<tr>
<th>Year</th>
<th>Capacity (kByte)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1980</td>
<td>2</td>
</tr>
<tr>
<td>1990</td>
<td>20</td>
</tr>
<tr>
<td>2000</td>
<td>2000</td>
</tr>
</tbody>
</table>
Situation (2)

– Target groups
  • user without computer science studies
  • "electrician"

– Target for the use of PLCs
  • replace contactor/relay
  • increase reliability
  • reduce costs

– Target for the PLC languages
  • to describe functions in known representations
    • ladder diagram (derived from the circuit diagram)
    • function plan (derived from the logic plan)

Standardization of the development of PLC systems
IEC 1131
DIN EN 611131
2.1 Automation computers

**Block diagram of the hardware structure of a PLC**

- Digital and analog input
- Central processing unit (CPU) and program memory
- RAM
- Timer clock
- Interface to programming device
- Digital and analog output
- Actuator signals
- Sensor signals

Internal bus connections.
Cyclic operation of a PLC

- PLC
  - Process image in output memory
  - Automation program
  - Process image in input memory

- Actuator signal
- Technical process in a technical system
- Sensor signal
Sequence of a cyclic program execution for a PLC

- Output process image from output memory
- Execute program
- Load process image in input memory
- Reaction time
- Process event
- Reaction on process event
- Start of 1st cycle
- Start of 2nd cycle
- Start of 3rd cycle
- Start of 4th cycle
# Features of PLC

<table>
<thead>
<tr>
<th>Advantage</th>
<th>simple programming through cyclic operational mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disadvantage</td>
<td>maximum reaction time on events in the technical process is equal to two program cycles</td>
</tr>
</tbody>
</table>

Program execution time:
- cycle time is not constant
- 1 ms per 1000 instructions
Microcontroller (One-chip-computer)

- highly integrated components
- employment for mass products
- composition of
  - standard micro processor
  - data memory / program memory
  - bus interfaces
  - process signal interfaces
- programming with development system
- short word length
- extremely low price, beginning at: 1-10 €
- good reliability and long life span
- high requirements regarding environmental conditions, temperature, humidity
Mikrocontroller 80C167

- 16-bit CPU
- 2 kBytes RAM
- 2kBytes RAM-extension
- ROM
- Watchdog Timer
- Clock
- Direct memory access
- Programmable digital input/outputs
- 10-bit AD-converter with multiplexor
- Serial input/output interfaces
- Pulse input/output
- Interrupt controller
- External data bus
- CAN bus interface
- 111 digital inputs/outputs
- 16 analog inputs
- 2 channels
- 32 inputs/outputs
- 56 interrupt inputs
- External data bus
- CAN bus

Simplified block diagram of a micro controller
2.1 Automation computers

**World market for microcontrollers**

- Manufacturers:
  Intel, Motorola, National Semiconductor, Toshiba, Infineon, Mitsubishi
- World-wide increasing market
- Increasing tendency to 16-/32bit microcontroller and specific digital signal processors

![Worldwide Microcontroller Shipments Forecast](chart.png)
Differentiation

- Microprocessor
  processor on a micro electronic chip

- Microcomputer
  all components on a micro electronic chip i.e., processor,
  memory, interface to periphery

- Microcontroller
  automation computer or an automation computer
  system on one chip
### Basic terms

- **CPU** = Central Processing Unit
- **RAM** = Random Access Memory
- **EPROM/PROM/ROM** = Erasable/Programmable/Read Only Memory
- **I/O** = parallel or serial input/output components
- **Counter/Timer** = generation of clock pulses
- **Interrupt Controller** = handling of hardware interrupts

**micro processor**

**data memory**

**fixed memory**

**process and data periphery**
**Industrial PC (IPC)**

- Pluggable circuit boards for connection of:
  - electrical process signals
  - optical process signals
  - bus systems
- Programming in high-level language
- Use of real-time operating systems
  - as single operating system
  - in addition to standard operating systems

**Operational areas of industrial PCs**

- Process visualization
- Process evaluation and monitoring
- Superordinated control tasks (control room tasks)
Environmental conditions

- Harsh environmental conditions
  - temperature variations
  - shocks and vibrations
  - dust and humidity
  - electrical or electromagnetic disturbances

- Protective measures in industrial PCs (IPC)
  - vibration absorbing disk drives
  - high quality of integrated components
  - special protecting case
### Protection forms of industrial PCs with IP index

<table>
<thead>
<tr>
<th>1st digit</th>
<th>Protection against solid objects</th>
<th>2nd digit</th>
<th>Protection against water impact</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No protection</td>
<td>0</td>
<td>No protection</td>
</tr>
<tr>
<td>1</td>
<td>Hand contact impossible (50mm objects)</td>
<td>1</td>
<td>Protection against vertical falling drops/ condensation</td>
</tr>
<tr>
<td>2</td>
<td>Finger contact impossible (12mm objects)</td>
<td>2</td>
<td>Protection against drops with a falling angle of 15°</td>
</tr>
<tr>
<td>3</td>
<td>Wire contact impossible (2.5mm objects)</td>
<td>3</td>
<td>Protection against rain fall up to 60°</td>
</tr>
<tr>
<td>4</td>
<td>Fine wire contact impossible (1.0mm objects)</td>
<td>4</td>
<td>Protection against spray from all directions</td>
</tr>
<tr>
<td>5</td>
<td>Protection against harmful dust</td>
<td>5</td>
<td>Protection against jets of water from all directions</td>
</tr>
<tr>
<td>6</td>
<td>Complete dust protection</td>
<td>6</td>
<td>Protection against water floods (during heavy sea)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td>Protection against water impact up to a depth of 1m</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td>Protection against persistent water impact for depths larger than 1m</td>
</tr>
</tbody>
</table>

**IP = Ingress Protection**
2.1 Automation computers

**Pure IPC solution**
- acquisition of process data
- execution of control program under real-time conditions
- interface to the user

**Advantages of a pure IPC-solution**
- better scalability of hardware
- many operating systems available
- wide spectrum of programming languages available
- open system for the integration of ready-made subsystems
2.1 Automation computers

**Structure of a pure IPC system**

- **Industrial PC**
- **Control panel**
- **Remote maintenance**
- **Modem**
- **Printer**
- **Field bus**
- **Modules for input and output of signals**
- **Sensors and actuators**
Structure of a PLC-IPC system (chalkboard writing)
2.1 Automation computers

**Process control system (PCS)**

- Distributed computer systems, connected via bus systems
- Interconnection with PLC computers
- Usage of prefabricated program modules developed by the Manufacturer of the PCS
- Configuration by operator

**Complete solutions from a single manufacturer**

- No compatibility problems
- Uniform process operation and monitoring
- High availability
- Defined responsibility
- Long lifetime
### Fields of application of process control systems

- Power plant automation
- Process engineering
- Building automation
- Production engineering

### Components of a PCS

- Operation and monitoring components
- Field level components
- Communication systems
- Engineering tool
Schematic structure

- operation master computer
- display and operation component (DOC)
- display and operation component (DOC)
- engineering station
- Redundant, manufacturer specific bus system
- field level component (FLC)
- field level component (FLC)
- field level component (FLC)
- field bus
- sensors and actuators
- field device

factory bus
2.1 Automation computers

- Central operation and monitoring
- Terminal bus (Ethernet)
- Local operation and monitoring
  - Operating interface
  - CM = connection module
- Server
- Central engineering system
- Fieldbus
  - PLC
  - Bus interface
  - Field bus
  - Actuator/sensor bus

CM = connection module
Operation and monitoring components

- Functions:
  - create/modify recipes/batch processes
  - change current values
  - communication with the process
  - process alarm messages and operating requirements
  - process visualization
  - interface to data base system for process data logging

- Realization:
  - PC, IPC, Workstation
  - Windows 95, Windows NT, Unix
Components at the field level

- In-house developments of the control system - manufacturer (proprietary solution)

- Setup
  - automation computer: PLC, IPC
  - distributed periphery
  - field devices: sensors, actuators
System communication

Bus systems
Operation level: standardized Ethernet bus
Process and field level: Field bus, e.g. Profibus, H1-Bus, Modbus, Interbus-S
Lowest field level: fast actuator-sensor buses
2.1 Automation computers

Engineering

- Tasks
  - Configuration
  - Programming
  - Maintenance

- Tools
  - Graphical tools (IEC 1131)
  - Standardized libraries with components
  - Powerful editors
## Control system manufacturers and their products

<table>
<thead>
<tr>
<th>Manufacturer</th>
<th>Name of the product</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABB</td>
<td>AdvantOCS</td>
<td>Company-specific field bus</td>
</tr>
<tr>
<td></td>
<td>AdvaSoft</td>
<td>For small systems</td>
</tr>
<tr>
<td></td>
<td>Procontrol P</td>
<td>Power plant automation</td>
</tr>
<tr>
<td>EB Hartmann &amp; Braun</td>
<td>Symphony</td>
<td>Supports connection to operation level, remote I/O</td>
</tr>
<tr>
<td></td>
<td></td>
<td>System automation for large systems</td>
</tr>
<tr>
<td></td>
<td>Contronic E</td>
<td>Power plant automation, for large installations</td>
</tr>
<tr>
<td></td>
<td>Contronic P</td>
<td>Process engineering, extension up to 12km</td>
</tr>
<tr>
<td>Foxboro-Eckardt</td>
<td>I/A Serie-System</td>
<td>Process engineering, field level with PCMCIA-technology</td>
</tr>
<tr>
<td>Siemens</td>
<td>SIMATIC PCS 7</td>
<td>Process engineering, user interface based on Windows 95 and Windows NT.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Extensive hardware supply. Field bus: Profibus. Connection to operation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>level possible.</td>
</tr>
<tr>
<td></td>
<td>Teleperm M</td>
<td>Process engineering, common old bus system (CS 275). Migration from</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Teleperm M to SIMATIC PCS 7 possible.</td>
</tr>
<tr>
<td></td>
<td>Teleperm XP</td>
<td>Power plant automation, open communication, extensive hardware supply.</td>
</tr>
<tr>
<td>Honeywell</td>
<td>PlantScape</td>
<td>Process engineering, open system, based on Windows NT, supports remote</td>
</tr>
<tr>
<td></td>
<td></td>
<td>I/O. Connection to operation level possible.</td>
</tr>
<tr>
<td></td>
<td>TDC 3000</td>
<td>Process engineering, has several process buses with different data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>transmission. MODBUS is supported.</td>
</tr>
</tbody>
</table>
§ 2  Automation Device Systems and Structures

2.1  Automation computers

2.2  Automation structures

2.3  Automation hierarchies

2.4  Distributed automation systems

2.5  Automation structures with redundancy
2.2 Automation structures

**Structure of the technical process**
- technical process as an entity  
  e.g.: drilling process with a drilling machine
- technical process consisting of sub-processes  
  e.g.: manufacturing of a gear

**Structure of automation devices**
- locally concentrated
- locally distributed

**Functional structure of automation systems**
- allocation of automation functions on the automation devices
  - functionally centralized
  - functionally decentralized
Locally concentrated automation devices

- measured-value transformer
- correcting elements
- sub-distributor
- jumper board, galvanic disconnector, automation device
- display
- monitoring signals
- multi-color screen for process management

Diagram showing the flow of signals from the field to the control room, including devices such as a measured-value transformer, correcting elements, sub-distributor, jumper board, galvanic disconnector, automation device, display, and multi-color screen for process management.
2.2 Automation structures

Locally distributed automation devices

CM: connection module for sensors and actuators

Multi-color screen for process management

Display

Master computer

PLC

Field bus

Plant bus

Control room
Functionally centralized automation structure

- Central automation computer
- Sub-processes 1, 2, ..., n
- Technical process in a technical system
2.2 Automation structures

**Functionally decentralized automation structure**

![Diagram showing a decentralized automation structure with multiple automation computers and subprocesses connected by arrows indicating data flow.]

- **automation computer 1**
- **automation computer 2**
- **automation computer n**
- **sub-process 1**
- **sub-process 2**
- **sub-process n**

The diagram illustrates how independent automation systems can be connected to handle different subprocesses, emphasizing decentralization and modular design.
## Different combinations of automation structures

<table>
<thead>
<tr>
<th>Functional structure</th>
<th>Technical process seen as an entity (centralized process structure)</th>
<th>Technical process segmented into sub-processes (decentralized process structure)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>locally centralized deployment of automation devices</td>
<td>locally decentralized deployment of automation devices</td>
</tr>
<tr>
<td>Functionally centralized automation structure</td>
<td>CCC</td>
<td>CDC</td>
</tr>
<tr>
<td>Functionally decentralized automation structure</td>
<td>CCD</td>
<td>CDD</td>
</tr>
</tbody>
</table>

**Notation of the different structures:** C = centralized / D = decentralized

1st digit  2nd digit  3rd digit

- Functional automation structure
- Local structure of the automation devices
- Structure of the technical process
**Criteria for a comparison of automation structure features**

- **Costs** for devices, cabling, software, maintenance and service
- **Availability** of parts in case of hardware failure or software faults
- **Flexibility** in case of modifications
- **Coordination** of sub processes and **optimization** of the overall process
- **Operability/usability**
Comparison of purchasing costs

- Functionally centralized structure causes no additional costs when upgrading.
- Functionally decentralized structure incurs increased purchasing costs with increasing number of sub-processes and/or automation functions.

Central structure with certain reserve causes no additional costs when upgrading.
Comparison of reliability during operation of technical processes

- Functionally decentralized, process malfunction with two single failures
- Functionally decentralized, process malfunction with one single failure
- Generally weak coupling
  - No complete failure if two or more automation units fail
  - Operational reliability is higher in the case of a functionally decentralized structure than the deployment of a central process computer
Decentralized structure

+ flexibility in case of modifications
+ coordination of sub-processes
+ optimization of the overall process
- additional effort for the communication between the individual automation units
0 operability and user-friendliness
+ localization of malfunctions
+ higher transparency
### Evaluation of automation structures regarding the criteria

<table>
<thead>
<tr>
<th>Structure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCC</td>
<td>Typical for the automation of small devices</td>
</tr>
<tr>
<td>CDC</td>
<td>Lower cabling costs as in CCC</td>
</tr>
<tr>
<td>DCC</td>
<td>Unfavorable regarding availability, maintenance, cabling costs</td>
</tr>
<tr>
<td>DDC</td>
<td>Unfavorable regarding availability and flexibility</td>
</tr>
<tr>
<td>CCD</td>
<td>Favorable regarding maintenance and flexibility, unfavorable regarding cabling</td>
</tr>
<tr>
<td>CDD</td>
<td>Favorable regarding flexibility, availability, cabling and transparency</td>
</tr>
<tr>
<td>DCD</td>
<td>Favorable regarding availability, maintenance, unfavorable regarding cabling costs</td>
</tr>
<tr>
<td>DDD</td>
<td>Favorable regarding flexibility, availability, cabling and transparency</td>
</tr>
</tbody>
</table>

---

**Functional structure**

- As decentralized as possible, as centralized as necessary

---

**Product automation**

**Plant automation**

**Automotive electronics**

© 2004 IAS, Universität Stuttgart
## § 2 Automation Device Systems and Structures

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Automation computers</td>
</tr>
<tr>
<td>2.2</td>
<td>Centralized and decentralized structures</td>
</tr>
<tr>
<td><strong>2.3</strong></td>
<td><strong>Automation hierarchies</strong></td>
</tr>
<tr>
<td>2.4</td>
<td>Distributed automation systems</td>
</tr>
<tr>
<td>2.5</td>
<td>Automation structures with redundancy</td>
</tr>
</tbody>
</table>
Combination of centralized and decentralized structures

- Introducing a hierarchy of automation units

- Combination of centralized and decentralized structures

- Requirements on availability

- Complexity of operational tasks

- Central master level

- Coordination level

- Process level

- Technical Process (overall process)
Allocation of automation functions to the process management levels

- decentralized automation units perform tasks of the field level, high requirements on availability

- coordination units perform the automation functions of the process level, as well as the coordination of sub-processes, optimization, process monitoring and safety functions

- master units perform the tasks of the operational level
2.3 Automation Hierarchies

Requirements on availability and processing capacity

- Protection, safety monitoring
- Locking
- Single controls
- Regulations
- Management regulation
- Management control
- Performance value calculation
- Optimization

Field level | Coordination level | Master level

© 2004 IAS, Universität Stuttgart
2.3 Automation Hierarchies

**Prerequisites for the realization of an automation hierarchy**

- segmentation of the technical process into sub-processes
- deployment of intelligent automation units
- communication system between automation units

Depending on the size of the company and the extent of the technical process, sub-levels can be added or levels can be joined.
Example

Realization of an automation hierarchy with a bus-oriented distributed process computer system
## § 2 Automation Device Systems and Structures

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Automation computers</td>
</tr>
<tr>
<td>2.2</td>
<td>Centralized and decentralized structures</td>
</tr>
<tr>
<td>2.3</td>
<td>Automation hierarchies</td>
</tr>
<tr>
<td>2.4</td>
<td><strong>Distributed automation systems</strong></td>
</tr>
<tr>
<td>2.5</td>
<td>Automation structures with redundancy</td>
</tr>
<tr>
<td>Objectives when using distributed automation systems</td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>High reliability through fault tolerance</strong></td>
<td></td>
</tr>
<tr>
<td>• failure of a decentralized computer does not lead to an overall failure</td>
<td></td>
</tr>
<tr>
<td>• isolation of the fault by means of reconfiguration</td>
<td></td>
</tr>
<tr>
<td><strong>Increase of availability through rapid maintenance and service</strong></td>
<td></td>
</tr>
<tr>
<td>• mutual control with fault diagnostics</td>
<td></td>
</tr>
<tr>
<td><strong>Mutual support during peak load times</strong></td>
<td></td>
</tr>
<tr>
<td>• automatic adaptation of the task distribution</td>
<td></td>
</tr>
<tr>
<td>• reduction of the standby capacity of the individual units</td>
<td></td>
</tr>
<tr>
<td><strong>Simple upgradeability</strong></td>
<td></td>
</tr>
</tbody>
</table>
Realization when using distributed automation systems

- Connection of decentralized automation units with a superordinated computer via a communication system

no hierarchy

Difference to the automation hierarchy

- In automation hierarchy, units only communicate with the level directly above it
- Only the process-related information tasks are perceived as decentralized
Selection of a communication system (1)

- Low cabling costs
  - essential costs: more cable cause more interferences in the wires

- Standardized interfaces regarding
  - plugs, cables (mechanical)
  - voltage level (electrical)
  - transmission protocol (logical)

- Flexibility in case of modifications

- Low requirements on the communication partner
  - demand on memory size
  - demand on computation power

- High availability and reliability
Selection of a communication system (2)

- Error free transfer of information
  - utilization of test bits
  - acknowledgment of correct reception
- Achievement of high data transmission rates
- Short reaction times on transmission requests
- Coupling of various communication partners

Problem:
Definition of priorities in the realization of the partially contradicting criteria

e.g.: high availability and reliability through redundant bus systems can cause high cabling costs
Basic topologies of communication

a) star topology

b) ring topology

c) net topology

d) bus topology

CAU = centralized automation unit
DAU = decentralized automation unit
Basic topologies of communication

**star topology**
failure of the central unit causes failure of the communication system

**ring topology**
each unit can only transmit messages to its direct neighbors

**net topology**
parallel information transmission, short reaction time, many interfaces, high cabling costs

**bus topology**
only one participant at a time is able to send, simultaneous information reception from all participants
### Parallel bus
- addresses, data and control signals are transmitted parallel
- bundles of wires

**distance:** up to 20 m

### Serial bus
- bits of a message are transmitted one after another
- transmission time is longer than that of parallel bus
- lower cabling costs
- increase of reliability
- flexibility regarding the transmission protocol

**distance:** 20 m up to 15 km
2.4 Distributed automation systems

Types of communication systems

- Open communication system
  OSI = Open-System-Interconnection

- Manufacturer-specific communication system (proprietary systems)
  CSI = Closed-System-Interconnection
## Tasks of ISO/OSI-layers

<table>
<thead>
<tr>
<th>Layer</th>
<th>Layer Name</th>
<th>Functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Application layer</td>
<td>- basic services</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- standard applications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- application-specific</td>
</tr>
<tr>
<td>6</td>
<td>Presentation layer</td>
<td>- language adjustments</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(e.g., between ASCII &amp; EBCDIC)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data encoding</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data decoding</td>
</tr>
<tr>
<td>5</td>
<td>Session layer</td>
<td>- establish</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- control</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- stop communication</td>
</tr>
<tr>
<td>4</td>
<td>Transport layer</td>
<td>- route parallelism</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- packet retransmission</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- packet sorting</td>
</tr>
<tr>
<td>3</td>
<td>Network layer</td>
<td>- net protocols</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data addressing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- data switching</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- routing</td>
</tr>
<tr>
<td>2</td>
<td>Data link layer</td>
<td>- error detection/ handling</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- access mode</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- synchronization</td>
</tr>
<tr>
<td>1</td>
<td>Physical layer</td>
<td>- transmission medium</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- type of coding</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- interface</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- topology</td>
</tr>
</tbody>
</table>
§ 2 Automation Device Systems and Structures

2.1 Automation computers
2.2 Centralized and decentralized structures
2.3 Automation hierarchies
2.4 Distributed automation systems
2.5 Automation structures with redundancy
2.5 Automation structures with redundancy

**Types of redundancy**

*Always: operating personnel monitoring parallel to process computer!*

- **Hardware redundancy**
  - redundant hardware

- **Software redundancy**
  - redundant software

- **Measured value redundancy**
  - redundant measured value
  - dependent measured value

- **Time redundancy**
  - multiple inquiry of the same measured value in certain intervals

*e.g.: velocity, acceleration*

Hardware and software redundancy implies a bigger effort

- higher availability
- safety-relevant systems
Objectives when using fault-tolerant structures

Design the systems in such a way that they are able to function as a whole even if faults occur in individual components.

Levels of fault-tolerance

- complete fault-tolerance  
  fail operational

- reduced operational capacity
  fail soft, graceful degradation

- transition to a safe state
  fail-safe
The principle of fault-tolerance

Construct a system with redundant modules (hardware and software) in order to maintain a functioning system in case errors occur.

Types of redundancy

- **Static redundancy**
  - all redundant modules are permanently in operation

- **Dynamic redundancy**
  - redundant modules are only used after a failure occurs
  - blind redundancy
    - redundant modules do not act in fault-free cases
  - function-participating redundancy
    - redundant modules run stand-by-functions in fault-free cases
Hardware redundancy

- Goal:
  Detection of hardware failures

- Operation principle:
  m-of-n-redundancy
  • majority ruling
  • no faults, until multiple defects occur

- Realization of redundancy
  • double computer structures
  • triple computer structures
Double computer structures with static redundancy

- Input signal (e.g., measured value)
- Computer 1
- Computer 2
- Comparator 2-out-of-2
- Output signal (e.g., manipulated variable)
- Alarm signal
Double computer structures with dynamic blind redundancy

- Input signal
- Main computer
- Stand-by computer
- Output signals
- Alarm message
- M = Monitoring Program
Double computer structures with dynamic function-participating redundancy

\[ M = \text{Monitoring Program} \]
Triple computer structures with static redundancy

input signals

computer 1

computer 2

computer 3

comparator 2-out-of-3

output signals
Software redundancy

- Objective:
  detection of errors in software

- Starting point:
  software has errors

Redundancy measures for software existence of the same software makes no sense, failure of software is not the problem

- heterogeneous structure of program components
- the same input data must lead to the same results
Diversity software

Diversity = Heterogeneity of software with identical functionality
– independent development teams solve the same problem
– intentional development of different strategies, algorithms and software structures

Application and execution of diverse software components

– Redundant software alternatives are executed one after another and are compared with the help of a voter (not for real-time systems with high requirements on timeliness)
– simultaneous execution of redundant software components on redundant multi-computer systems
– cyclic alternation of diverse components

Comparison difficult
– Two algorithms with different processing times
– Both results can be correct even though their values might be different
Question referring to Chapter 2.4

The individual modules of a PLC are connected via a manufacturer-specific bus. For the communication between field devices a field bus is often used.

Explain at this example the difference between an “open“ and a “proprietary“ communication system.

Answer

The PLC bus system is a proprietary communication system. This means, it is a manufacturer-specific system where only devices made by the same manufacturer can communicate with each other.

A field-bus system is an open communication system. Devices of different manufacturers can be connected with this system.
Question referring to Chapter 2.4

There are some different topologies for a communication system. Which of the following statements do you agree?

- the bus topology causes the lowest cable expense
- the bus topology is faster than the net topology
- a net topology is able to broadcast a larger amount of data in parallel than a bus topology
- a bus topology has a shorter reaction time than a net topology
- it is easier to expand a net topology than a star topology
- a bus topology is only suitable for decentralized systems

Answer
Question referring to Chapter 2.5

In aircraft’s important systems, variables are calculated more than once. Which kind of system would you use for this application: a double or a triple computer system with static redundancy?

A double computer system with static redundancy is not fault-tolerant. In case of a failure the system is shut down. Therefore, a triple computer system has to be used. Because of the 2-out-of-3 decision, this type of system is fault-tolerant.

Answer
Crosswords to Chapter 2
Crosswords to Chapter 2

Across
2 Communication medium (3)
3 Heterogeneity of software with identical functionality (9)
7 Physically distributed (13)
9 One chip computer (15)

Down
1 Preservation of the functionality despite occurrence of errors. (5,9)
4 Abbreviation for computers specially designed for usage in industrial fields. (3)
5 Duplication of hardware or software elements with the same functionality. (10)
6 Simultaneous transmission of data on several lines. (8)
8 Time impulse generator (5)